Specification:

Replace the first full paragraph on page 3 with the following rewritten paragraph:

As digital circuits become more and more complex while integrated circuit technologies grow, equivalence checking methods and tautology checking methods often fail to handle the complex Boolean functions used to represent digital circuits. The failures are either due to unreasonably long run times or due to requiring unreasonably large amounts of computer memory. These methods (such as U.S. patent No. 5,243,538 and others using BDDs) give meaningful conclusions only at the end, and therefore they give no meaningful conclusions at all if they fail before reaching the end.

Replace the second full paragraph on page 3 with the following rewritten paragraph:

A way to avoid such failures is to check equivalence or tautology for only some of all combinations of the variable values. This provides meaningful conclusions before the end of checking for all combinations of the variable values. This is used widely in the industry because each of the ignored combinations is often similar enough to a combination involved in the checking or some combinations are less critical than others. This is normally performed as logic simulation for one combination at a time. When the same combinations are involved, logic simulation is

A2 usually much less efficient than equivalence checking methods or tautology checking methods.

Replace the first full paragraph on page 6 with the following rewritten paragraph:

A3 A cube, as well known in the art, is a subset of the input space where some input variables are substituted with Boolean constant 1 and some other input variables are substituted with Boolean constant 0, which is called the substitution requirements provided by the cube in this specification. The count of points in a cube must be 1, 2, 4, or any other power of 2 because zero or more input variables are completely free to take any values (as they are not substituted with either Boolean constant according to the substitution requirements provided by the cube). A cube of 0 point does not occur in this context. A Boolean function is a conditional tautology with a cube as the given subset of the input space if the Boolean function is simplified to Boolean constant 1 after the input variables in the Boolean function are substituted with these 1's or 0's according to the substitution requirements provided by the cube ~~the cube's definition~~.

Replace the last paragraph on page 6 with the following rewritten paragraph:

A4 Conditional tautology checking can be performed in a recursive process as illustrated in FIG. 1. The process starts with a step **100** receiving a Boolean function and a given subset of the input space. A step **110** is performed next to determine whether to go to a step **120** or a step **150**. It must go to step **150** if the given subset of the input space is not a cube. It may go to either step **120** or step **150** under other conditions, but it must go to step **120** if the given subset of the input space includes only one point. The Boolean function is simplified in step **120** with substituting input variables in the Boolean function with Boolean constants according to the substitution requirements provided by the cube (the given subset of the input space) ~~the cube's definition~~. A step **130** is performed after step **120** to check the simplification result before going to step **150** or another step **140**. Step **150** is performed next if the simplification result is not a Boolean constant. Otherwise step **140** is performed next to look at the Boolean constant resulted from the simplification. The positive conclusion is given in a step **180** after step **140** if the simplification result is Boolean constant 1. The negative conclusion is given in a step **190** after step **140** if the simplification result is Boolean constant 0. The negative conclusion may include at least one counter example indicating the substitution of the input variables with the Boolean constants. A4

Replace the last paragraph on page 8 with the following ~~re~~written paragraph:

Conditional tautology checking can also be performed in the following iterative process illustrated in FIG. **2**. The iterative process starts with a step **200** receiving a Boolean function and a given subset of the input space. A step **210** is performed next to divide the given subset of the input space into two subsets. One is called a cube subset and the other is called a tail subset. The cube subset is a cube in the A5

A5

input space. If the given subset is a cube itself and it is chosen to be the cube subset, the tail subset is empty. A step **220** is performed after step **210** to simplify the Boolean function with substituting input variables in the Boolean function with Boolean constants according to the substitution requirements provided by the cube (the given subset of the input space) ~~the cube's definition~~. The simplification result can be used to replace the original Boolean function in all following steps and iterations if the tail subset is empty. Otherwise the original Boolean function is preserved for all following steps and iterations, and the simplification result can be used temporarily as the Boolean function to ~~simply~~ be simplified again in the next round, as described below, ~~only when the simplification result is not~~ unless any further simplification is not needed because the simplification result is already a Boolean constant.

Replace the second full paragraph on page 10 with the following rewritten paragraph:

A6

Given an ordering of all the input variables, each point of the input space can be represented as a binary integer. Each bit of the binary integer represents the value of each input variable. The total count of these binary integers is the Nth power of two where N is the number of input variables. The lower bound of these binary integers is zero. A range of these binary integers represents a subset of the input space. A range has a lower bound and an upper bound, and it includes all binary integers between the two bounds (i.e. all binary integers that are greater than

A4

or equal to the lower bound and smaller than or equal to the upper bound). Any subset of the input space can be represented as one or more ranges of binary integers jointly (i. e. the subset is the union of the one or more subsets that are represented as the ranges) though certain subset of the input space may only be represented as multiple ranges of binary integers.

Replace the last paragraph on page 10 with the following rewritten paragraph:

A7

A range of these binary integers represents a cube if the only difference between the range's upper bound and the range's lower bound is at all the rightmost bits (~~i.e. the~~ specifically, the rightmost bits of the range's lower bound, which have to be all ~~if they are~~ 0's, all correspond to 1's in the rightmost bits of the range's upper bound, respectively). A range from ~~10100 to 10101~~ 10000 to 10111 represents a cube. A range from 10100 to 10110 does not represent a cube. A range from 10010 to 10111 does not represent a cube. A range of any single binary integer represents a cube. The range of all N-bit binary integers represents a cube.

Replace the second full paragraph on page 12 with the following rewritten paragraph:

A8

More generally, checking a conditional tautology can be performed as checking several smaller conditional tautologies concurrently if the division is decided in advance. Then each of the smaller conditional tautologies can be checked with dynamic divisions if the given subset of the input space of the smaller conditional tautology can be represented as a range of binary integers. Checking each smaller conditional tautology can start from either end of the range of binary integers representing the given subset of the input space of this smaller conditional tautology. It can start from different ends for checking different ones of these smaller conditional tautologies. If desired, the iterative process and the recursive process can be mixed in checking a conditional tautology. Different processes can be used either at different levels of the divisions or for different smaller subsets of the input space from the same division. When the (given or smaller) subset of the input space is a cube, any other tautology checking method ~~other than~~ in addition to simplification can also be applied after constant substitution.